TITLE OF THE INVENTION

Generalized convolutional interleaver/de-interleaver

CROSS REFERENCE TO RELATED PATENTS/PATENT APPLICATIONS

The present U.S. Utility Patent Application is a continuation of the following U.S. Utility Patent Application which is hereby incorporated herein by reference in its entirety and made part of the present U.S. Utility Patent Application for all purposes:

1. U.S. Utility Patent Application Serial No. 10/325,525, entitled "Generalized convolutional interleaver/de-interleaver," (Attorney Docket No. BP1199CON), filed December 19, 2002 (12/19/2002), pending.

The U.S. Utility Patent Application Serial No. 10/325,525 is a continuation of the following U.S. Utility Patent Application which is hereby incorporated herein by reference in its entirety and made part of the present U.S. Utility Patent Application for all purposes:

1. U.S. Utility Patent Application Serial No. 09/430,456, entitled "Generalized convolutional interleaver/de-interleaver," (Attorney Docket No. BP1199), filed on October 29, 1999 (10/29/1999), which issued as U.S. Patent No. 6,546,520 on April 8, 2003 (04/08/2003).

The U.S. Utility Patent Application Serial No. 09/430,456 claims priority pursuant to 35 U.S.C. § 119(e) to the following U.S. Provisional Patent Applications which is hereby incorporated herein by reference in its entirety and made part of the present U.S. Utility Patent Application for all purposes:

1. U.S. Provisional Patent Application Serial No. 60/106,482, entitled "Efficient convolutional interleavers/de-interleavers," filed October 30, 1998 (10/30/1998), pending.

BACKGROUND OF THE INVENTION

TECHNICAL FIELD OF THE INVENTION

The invention relates to an apparatus and method for convolutional interleaving/deinterleaving.

DESCRIPTION OF RELATED ART

Present digital communication channels are experiencing greatly increased demands, which lead to errors in the data being transmitted in that channel. The error correcting codes in common use are very good at detecting and correcting isolated bit errors which occur in a communication channel. However, typical error correction and detection codes are insufficient in a channel which is subject to burst-type errors, i.e., errors which will affect a large number of bits of data at a time.

Convolutional interleaving and de-interleaving techniques on either end of the channel transmission path are used to interleave the data stream, so that the effects of burst errors become distributed when the data stream is de-interleaved, and do not overwhelm the error correcting and detecting codes.

A balance often must be maintained among considerations such as, for example, the physical amount of memory used to realize the interleaver/de-interleaver, the total amount of device "real estate" available to the device designer, the device performance, device flexibility and dynamic re-programmability, and the simplicity of the device design and implementation. In applications where the importance of spatial efficiency is less important, the interleaver/de-interleaver can be implemented using an arbitrary number of memory cells, provided the requisite device performance characteristics are met. Often, interleavers/de-interleavers are realized using distinct designs and implementations, which cannot be reconfigured dynamically to satisfy, for example, the demands of a different environment requiring the use of a different type of interleaver/de-interleaver.

The tension of this balance is most prominent in single-chip signal processing device implementations, where spatial efficiency can become a crucial consideration. There is a need for efficient implementations of certain types of interleavers/de-interleavers, including, for example, a Ramsey Type II device, have not been demonstrated. Furthermore, there is a need for an interleaver/de-interleaver that can be dynamically reconfigurable among the different types of devices, for example, Ramsey I, Ramsey II, Ramsey III, and Ramsey IV.

BRIEF SUMMARY OF THE INVENTION

The invention provides a memory-efficient convolutional interleaver/de-interleaver which includes a memory array, a write commutator, and a read commutator; wherein the commutators perform their respective write and read operations relative to a preselected memory cell after a predetermined delay. The delay is chosen using a technique, such as a modulo-based technique, such that an efficient implementation of a Ramsey Type-II interleaver, and a Ramsey Type-III is realized.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

Figure 1 is an exemplary illustration of an embodiment of the present invention, in the form of a Ramsey Type-II interleaver.

Figure 2 is an exemplary illustration of an embodiment of the present invention, in the form of a Ramsey Type-III interleaver.

DETAILED DESCRIPTION OF THE INVENTION

Although convolutional interleavers/de-interleavers are well known in the art, the design of memory-efficient devices approaching the theoretical minimum number of memory locations has been elusive, particularly interleavers/de-interleavers of the Ramsey II-type. The invention herein provides a generalized memory-efficient interleaver/de-interleaver that can be reconfigured to selectively operate as distinct devices, such as a Ramsey I-, Ramsey II-, Ramsey, III-, or Ramsey IV-type interleaver/de-interleaver. The implementation of the Ramsey-II type of device according to the present invention is substantially efficient, to the extent where such a design can approach a theoretical minimum number of memory cells.

The concept of convolutional interleaving was first introduced by J. L. Ramsey and G. D. Forney in around 1970. See, for example, J. L. Ramsey, "Realization of Optimum Interleavers," IEEE Information Theory, Vol. IT-16, Number 3, May 1970, pp. 338-345; and G. D. Forney, "Burst-Correcting Codes for the Classic Bursty Channel," IEEE Trans. Communication Technology, Vol. COM-19, Oct. 1971, pp. 772-781. When convolutional interleaving is used, the total memory requirements can theoretically be reduced to approximately N*d. The reduction in memory results in a memory requirement which is one-fourth the requirement of block interleaving. At the same time, the overall latency is reduced by up to approximately the same level (i.e., N*d). This is approximately half of the total latency of a block interleaver.

The implementations of convolutional interleaving described by Ramsey and Forney as well as others are capable of achieving the reduction of required memory by up to a factor of 4 through the use of 2*N separate delay lines. Each of the delay lines can be implemented by a RAM with its own address. However, when the number of symbols in a block of data N is relatively large, a correspondingly large number of separate RAMS are required in order to implement the interleaver. In order to improve the efficiency of the system, it would be desirable to consolidate these memories into a relatively small number of RAMS.

U.S. Pat. No. 4,559,625, entitled "Interleavers for Digital Communications," issued Dec. 17, 1985 to E. R. Berlekamp, et al. describes an interleaving system that requires only one RAM for the interleaver and one additional RAM for the de-interleaver.

However, in the described implementation, which is referred to as "helical" interleaving, the interleaving depth d is restricted to a value that is one greater or one less than the number of symbols in a block of data N. That is, D=N±l. Although the "helical" interleaver disclosed by Berlekamp, et al. works well in some applications, it is limited in that it cannot be used in applications which require different relationships between the interleaving depth and the number of symbols in a block of data.

More recently, J. T. Aslanis, et al. described a convolutional interleaving system that permits an arbitrary interleaving depth d wherein the only restriction on the interleaving depth d is that it must be co-primed with the number of symbols in a block of data N. The described system uses a single RAM implementation with a total memory requirement equivalent to 2*N*d. See generally, Aslanis et al. "An ADSL Proposal for Selectable Forward Error Correction with Convolutional Interleaving", TIEI.4/92-180, Aug. 20, 1992. It should be appreciated that although this system requires just half of the memory required by the block interleaver, it still requires an amount of memory which is approximately twice as high as the theoretical minimum.

Also, U.S. Pat. No. 5,764,649, entitled, "Efficient Address Generation For Convolutional Interleaving Using A Minimal Amount Of Memory," issued June 9, 1998 to PO Tong, describes an addressing scheme which uses a more reduced amount of memory in the interleaving and deinterleaving process, thereby achieving a significant savings in memory requirements. However, to realize this savings, a rather involved addressing scheme is required which involves generating several arrays which characterize the delays relating to each symbol, as well as the addresses for both the interleaver and the de-interleaver.

Each of the above-described references is incorporated herein in its respective entirety.

The present invention provides a generalized, memory-efficient, convolutional interleaver/de-interleaver, the configuration of which can be dynamically selectable. For the purposes of clarity, and due to the duality between an interleaver and a de-interleaver, an interleaver/de-interleaver will be called an interleaver, as will be understood by skilled artisans. Also known to skilled artisans, a Ramsey Type-I interleaver is the dual of a Ramsey Type-II interleaver, and a Ramsey Type-III interleaver is the dual of a Ramsey

Type-IV interleaver. Therefore, a Ramsey Type-I interleaver can be used as a de-interleaver when used in conjunction with a Ramsey Type-II interleaver.

In its most general form, an interleaver includes an array of memory cells, a write commutator and a read commutator. It is desirable for each of the commutators to have their positions relative to the previous position in memory cells updated, according to a preselected technique, at least one of which being a modulo-based technique. Where only one technique is a modulo-based, the other preselected technique can include updating the respective commutator by an offset of one or more positions, relative to the previous position in the memory cells.

Figure 1 illustrates an exemplary embodiment of a Ramsey Type II interleaver/de-interleaver. As shown in Figure 1, the present invention comprehends an interleaver 1, and a de-interleaver 2, that convey information through a data channel 3. Each of interleavers 1, 2 are shaped approximately like a triangular memory array. Interleaver 1 includes write commutator 4, read commutator 5, and plural rows 16, 17 of memory array 9. In general, each interleaver and de-interleaver has one write commutator and one read commutator. Interleaver 1 also includes row position pointer 11; it is desirable for each row 16, 17 in memory array 9 have a row position pointer 11 associated therewith. De-interleaver 2 includes write commutator 6, read commutator 7, plural rows 18 of memory array 8, and row position pointer 13.

In the invention herein, memory locations can be referenced by row and row position. Typically, the row position is selected by the read/write commutator; and the position within a given row is selected by the row position pointer.

In this example, commutators 4, 6 are write commutators; and commutators 5, 7 are read commutators. For the purposes of clarity, the following exemplary process will be described from the perspective of interleaver 1; based on this information, a skilled practitioner can readily extend the process to de-interleaver 2. It is desirable, but not necessary, that commutators 4, 5 be initialized to the top row of the memory array. A symbol S_1 can be written at a first time to a first row 16 pointed to by write commutator 4. Also, symbol S_2 can be read at a second time, from second row 17 which is pointed to by read commutator 5. At this point, both pointers can be updated according to at least one predetermined update technique. It is desirable that read commutator 5 (and write commutator 6) be updated using a predetermined modulo technique. It is most desirable that a commutator "wraps" back on the array in an appropriate manner when the end of an array is reached.

Commutator updates can be made using many schemes, one being:

$$\mathbf{CommutatorPos_{i+1} = (CommutatorPos_i + H)\ mod\ N}$$

where: rows are labeled $0 \le R < N$;

N is the total number of rows; and **H** can be either 1 or $\boldsymbol{K}$, where $\boldsymbol{K}$ is computed by solving the equation:

$$KD \bmod N \equiv 1$$

It will be apparent to those skilled in the art that the difference between the second time and first time is representative of a predetermined delay that, in turn represents a preselected number of symbols S_1. Each row 16, 17 of memory array 9 can have a row position pointer 11 associated therewith, by which, the memory location for symbol B_1 can be selected for a respective read or write operation. Initial access to row 16 can be performed to an arbitrary position within row 16. It also is desirable that a read operation update row position pointer 11.

Row position pointer 11 of a given row 16, 17 can be updated in conjunction with a read operation or a write operation. It is desirable to update row position pointer 11 in conjunction with a write operation when interleaving, and with a read operation when deinterleaving. Typically, read and write operations occur in pairs, with a read-after-write sequence being desirable. In such a case, row position pointer 11 can be post-incremented if it is updated in conjunction with a write operation, and pre-incremented if it is updated in conjunction with a read operation, with the reverse becoming a desirable sequence if the read operation precedes the write operation.

The row position increment operation can include any method of successively pointing to individual memory locations in a given row 16. One such method can include:

$$\mathbf{RowPos_{i+1} = RowPos_i + L\ mod\ RowSize}$$

where L is an integer such that:

$$\gcd(L, RowSize) = 1$$

Other patterns and schemes may be used as well, for example, any such method which selects the elements in a given row, in any order. It is desirable that, once all elements have been selected, the selection sequence repeats.

Prior to a read operation row position pointer 11 is incremented using a preselected technique, such as, for example, a modulo technique, with the last position of a given row incrementing back to the first location of a row. The read operation is then performed from the location indicated by row position pointer 11. Where a write operation is used for updating, row position pointer 11 can be updated using a preselected modulo technique. In general, the modulo technique according to the present invention comprehends moving the appropriate commutator by a {modulo(number of rows)} displacement.

In the case of interleaver 1, write commutator 4 can be incremented by a fixed amount, or offset, for example, one position, after each write operation. When the last row of array 9 is reached, the commutator position wraps back around to the starting row of array 9 such that the entire cycle is repeated. Currently, it is desirable for read commutator 5 to be updated by K positions after each read operation. After successive read/write operations, the last row of the array is passed, and the position of read commutator 5 is updated by K rows. Using the desired interleaver block length N and interleave depth D, and interleave increment offset K can be computed by solving the equation:

$$KD \bmod N \equiv 1$$

Where memory array 9 consists of N rows of memory cells, and the index, R, of a logical sequential arrangement of the memory cells satisfies:

$$0 \le R < N;$$

starting with the top row, the size of each row in the logical arrangement can be given by:

$$S = \left\lfloor \frac{(D-1)}{N} R \right\rfloor + 1$$

where $\lfloor \ \rfloor$ is the floor operator.

Similarly, the size of each row of the de-interleaver array can be given by:

$$U = \left\lfloor \frac{(D-1)}{N} (N-1-R) \right\rfloor + 1$$

As a result, the total number of memory locations, ***M***, required to implement either the interleaver or the de-interleaver can be determined by:

$$M = \frac{(N-1)D + \gcd(N, D-1) + 1}{2}$$

where N and D are arbitrarily selectable parameters, as long as

$$\gcd(N, D) = 1\,.$$

The operation of de-interleaver 2 is logically similar to that of interleaver 1 in that write commutator 6 of de-interleaver 2 performs write operations on memory array 8 using an increment offset K for the repositioning of commutator 6, in the manner that read commutator 5 of interleaver 1 performs read operations on memory array 9 using the offset K for the repositioning of commutator 6.

In the case where rows are sequentially indexed, it is apparent that the length of the rows so configured either remains the same or increases, as a result of the floor operator function. Similarly, the length of the rows in de-interleaver 2 will remain the same of decrease. There is no requirement that the logical configuration of the memory correspond with the physical configuration, so that it is possible to substitute one row for another row. However, it is desirable that the interchanged rows be of the same length.

Figure 2 illustrates a Ramsey Type-III interleaver 15 and a Ramsey Type-.IV device 16 as the corresponding de-interleaver. As with devices 1, 2, devices 15, 16 respectively employ write commutators 17, 20; read commutators 19, 22; and row position pointers 24, 25 to perform interleaving and deinterleaving in respective memory arrays 18, 21. Similar to the number of rows, N, in Figure 1, devices 15, 16 are designed to use I rows of memory cells. As with interleavers 1, 2, the operation of commutators 19, 20, and perhaps, row position pointers 24, 25, can employ a selected modulo-based technique. It is most desirable that each of the commutators 17 and 22, and 19 and 20, and the row position pointers, 24, 25, be synchronized.

The foregoing merely illustrates the principles of the invention, and it will thus be appreciated that those skilled in the art will be able to devise various alternative arrangements which, although not explicitly described herein, embody the principles of the invention within the spirit and scope of the following claims.